LISTING OF CLAIMS

This listing of claims replaces all prior versions and listings of claims in the application:

Please amend the claims as follows:

1. (currently amended) A processor, comprising:

at least one register file;

at least one execution unit coupled to the at least one register file;

at least one bypass circuit operatively coupled to said at least one register file and said at least one execution unit, said at least one bypass circuit capable of arbitrating access to said at least one register file; and[[,]]

a backing register file operatively coupled to said at least one register file, and where said backing register file is operationally and responsively coupled to at least one user-visible instruction, the user-visible instruction having registering windowing capability or having the capability to use the backing register file in a native mode, wherein the native mode is the capability to address each register of the backing register file by way of addresses or at random.

2. (currently amended): [[A]] The processor ~~as in~~ of claim 1, further comprising a plurality of register files and further comprising at least one execution unit operably connected to each register file of said plurality of register files, and where said backing register file is operably connected to each register file of said plurality of register files providing thereby the ability to transfer values from any designated location in any designated register file of said plurality of register files to any designated location in said backing register file, and from any designated location in said backing register file to any designated location in any designated register file of said plurality of register files.

3. (currently amended): [[A]] The processor ~~as in~~ of claim 1, further comprising a connection circuit having a first connection and a second connection, where said first connection is operably connected to said backing register file from the at least one register file and said second connection is operably connected to a main memory from the said backing register file.

4. (canceled).

5. (currently amended): A method for moving values from designated locations in designated register files to designated locations in a backing register file and values in designated locations in said backing register file to designated locations in designated register files comprising:

(a) identifying a backing register file instruction in a sequence of instructions, the sequence of instructions constituting one or more instruction streams, wherein the sequence of instructions has registering windowing capability or has the capability to use the backing register file in a native mode, wherein the native mode is the capability to address each register of the backing register file by way of addresses or at random;

(b) decoding said backing register file instruction, where if said backing file instruction is one of load-backing-register-file or load-register-file, making available addresses for specified numbers of locations in specified register files and an equal number of addresses for specified locations in said backing register file, where said number of addresses is at least one;

(c) reading values from each of said addresses in said specified register file and writing said values to said equal number of addresses in said backing register file as specified by said backing register file instruction, if said backing register file instruction is of type load-backing-register-file; and[[,]]

(d) reading values from each of said addresses specified in said backing register file and writing said values to said equal number of addresses in said specified register file, if said backing register file instruction is of type load-register-file.

6. (canceled).

7. (currently amended): ~~A computer readable medium containing a backing register file instruction~~. A computer readable medium containing an identifiable backing register file instruction for moving data between a memory and a backing register file, wherein the backing register file instruction has registering windowing capability or has the capability to use the backing register file in a native mode, wherein the native mode is the capability to address each register of the backing register file by way of addresses or at random.

8. (currently amended): [[A]] The computer readable medium ~~as in~~ of claim 7, further comprising the backing register file instruction for transferring register values between a register file and said backing register file.

9. (currently amended): [[A]] The computer readable medium ~~as in~~ of claim 7, further comprising the backing register file instruction for transferring values between main memory and said backing register file.

10. (currently amended): [[A]] The computer readable medium of claim 7, further comprising a ~~set of instructions for a singly linked list and a byte stream, wherein the singly linked list and the byte stream contain backing register instructions~~ structure for requesting instructions to be sent to the backing register file, wherein the structure is a linked list or a set of fields of specified length, the specified length being a plurality of bytes.

11. (currently amended): [[A]] The processor of claim 1, further comprising the backing register file capable of being explicitly used by programs at all privilege levels.

12. (currently amended): A method of accessing a backing register file comprising:

identifying a backing register file instruction in an instruction stream, the identifying being the identification of an instruction for registering widowing capability or for using the backing register file in a native mode, wherein the native mode is the capability to address each register of the backing register file via addresses or at random;

switching modes to access a backing register file; and

moving values between a main memory and a backing register file.

13. (currently amended): [[A]] The method of claim 12, ~~further~~ wherein ~~one~~ a first mode emulates a legacy software, the emulation capable of permitting backward compatibility.

14. (new) The method of claim 12, wherein identifying the backing register file instruction further includes corresponding the backing register file instruction to a program from which the backing register file instruction originates.

15. (new) The method of claim 12, wherein identifying the backing register file instruction further includes allowing full access to the backing register file after the identification of the backing register instruction.

16. (new) The method of claim 15, wherein allowing full access further includes using an extended instruction set or a standard instruction of a new processor.

17. (new) The method of claim 15, wherein allowing full access further includes storing data in a structure, wherein the structure is a linked list or a byte stream.

18. (new): The method of claim 12, wherein a second mode is a native mode, wherein the native mode is the capability to address each register of the backing register file via addresses or at random.